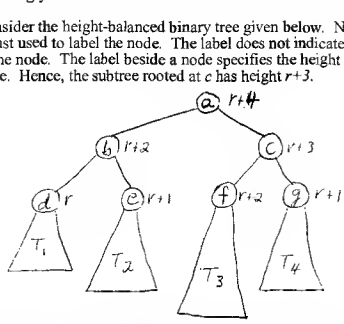
CMPT 250
Midterm 2

50 minutes | Closed Book | March 13, 2002

Marks

The exam is out of 50 and you have 50 minutes to complete the exam. Budget your time accordingly.

7 1. Consider the height-balanced binary tree given below. Note that the letter inside a node is just used to label the node. The label does not indicate anything about the value stored in the node. The label beside a node specifies the height of the subtree rooted at the node. Hence, the subtree rooted at c has height $r+3$.



(a) For each of the subtrees T_1, T_2, T_3, and T_4, suppose a deletion is done in the subtree that decreases the height of the subtree by 1. Does the height decrease result in the need for a rotation above it? If so, what is the root of the subtree that is rotated, and is it a left rotation or right rotation? If it is possible to tell whether it is a single or double rotation, specify which it is.

Note: answer the above questions four times, once for each of the four subtrees T_1, T_2, T_3, and T_4.

(b) For one of the rotations of part (a), show the tree that results from the rotation. Include the labels in the nodes, and label each node with the height of its subtree. If a subtree needs to be split into 2 subtrees in order to do the rotation, assume h and i are the labels in the two nodes that are the roots of the 2 subtrees formed.

10 2. Prove by induction that in a non-empty complete binary tree, the number of leaves is 1 more than the number of interior nodes. A leaf has no children, and an interior node has at least one child. A complete binary tree has all leaves on the same level, and all interior nodes have two children. Some examples of complete binary trees are

10 3. Give a collection of test situations for an insertion routine into an open-address hash table with linear probing.

8 4. Suppose that there is a large collection of (key, item) pairs to be stored in a dictionary, where the key is a 7 digit integer. The following data structures are being considered for the implementation of the dictionary

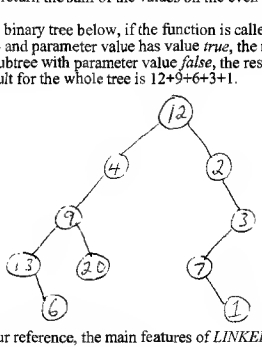
- two-three tree
- weight-balanced tree
- trie tree
- chained hash table

For each of these 4 implementations, into which of the following categories does it belong? Briefly explain why it fits in the category.

(a) Operations of insertion, searching and deletion usually take constant time, but in some situations, they might take linear time.
(b) Operations of insertion, searching and deletion take logarithmic time or less, never linear time.
(c) Operations of insertion, searching and deletion always take constant time.
(d) Doesn't fit in any of the above categories.

15 5. Give the Eiffel code for a INTEGER function called *alternating_sum* to be added into the class *LINKED_SIMPLE_TREE_UOS* [INTEGER]. The function *alternating_sum* has one parameter of type BOOLEAN. If the BOOLEAN parameter has the value *true*, then the function should return the sum of the values on the odd levels of the tree, i.e., sum of the value on level 1 (the root value), the values on level 3 (the grandchildren of the root), on level 5, etc. If the BOOLEAN parameter has the value *false*, the function should return the sum of the values on the even levels of the tree.

For the binary tree below, if the function is called on the subtree rooted at the node with value 4 and parameter value has value *true*, the result is 4+13+20. If it is called on the same subtree with parameter value *false*, the result is 9+6. With parameter value true, the result for the whole tree is 12+9+6+3+1.



For your reference, the main features of *LINKED_SIMPLE_TREE_UOS*[G] are:

is_empty : BOOLEAN	is_full : BOOLEAN
root_item : G	make
root_left_subtree : like Current	root_right_subtree : like Current
initialize (lt : like Current; x : G; rt : like Current)	
out : STRING	

Total 50 The end